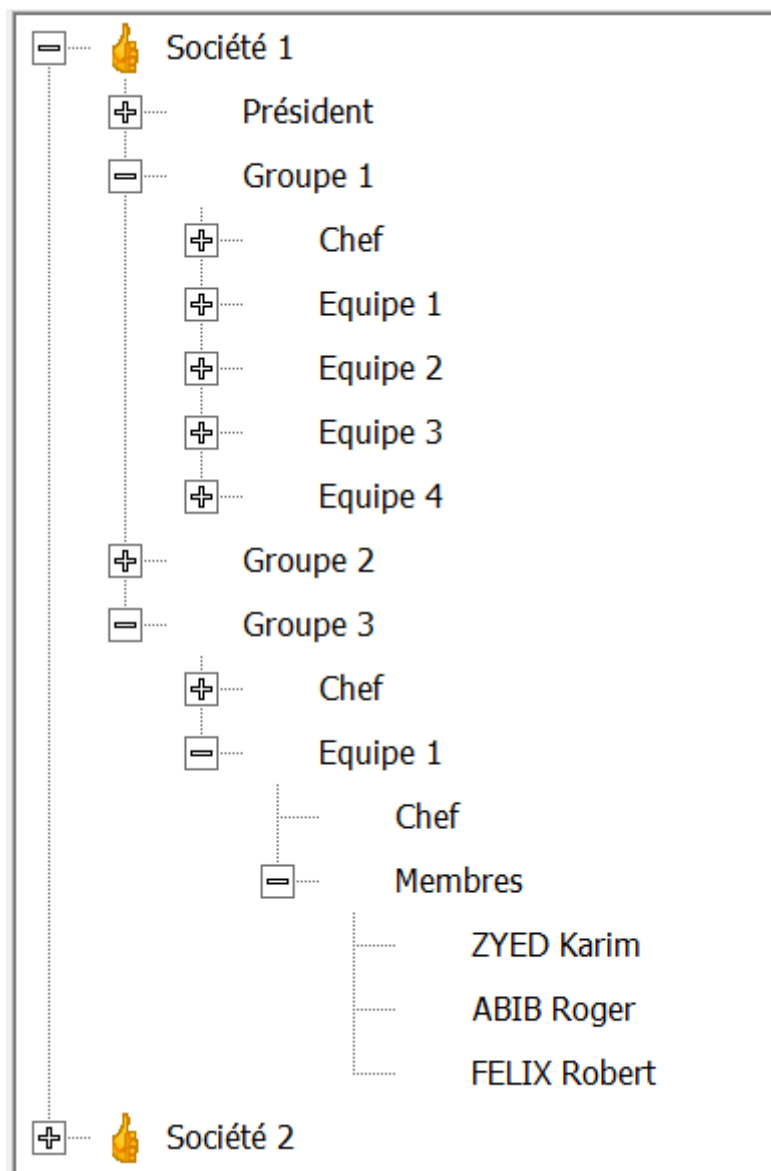


COURS TreeView VBA

Ce cours est une présentation rapide pour un premier aperçu de l'objet TreeView.

Il ne concerne que le TreeView dans un Userform.



SOMMAIRE

- **Présentation du TreeView.**
 - [Les Paramètres du TreeView.](#)
- [Ajouter le TreeView dans la boîte à outils.](#)
- **Remplissage du TreeView.**
 - [La Préparation.](#)
 - [Le Fonctionnement.](#)
 - [Le Programme.](#)
 - [Ajouter une Image sur un Nœud](#)
 - [Ajouter ImageList à la boîte à outils](#)
 - [Configurer l'ImageList](#)
 - [L'intégrer dans le Nœud](#)
- [Récupérer les informations d'un Noeud](#)
 - [Un Nœud Sélectionné](#)
 - [Un Nœud Coché](#)
 - [Globalement](#)

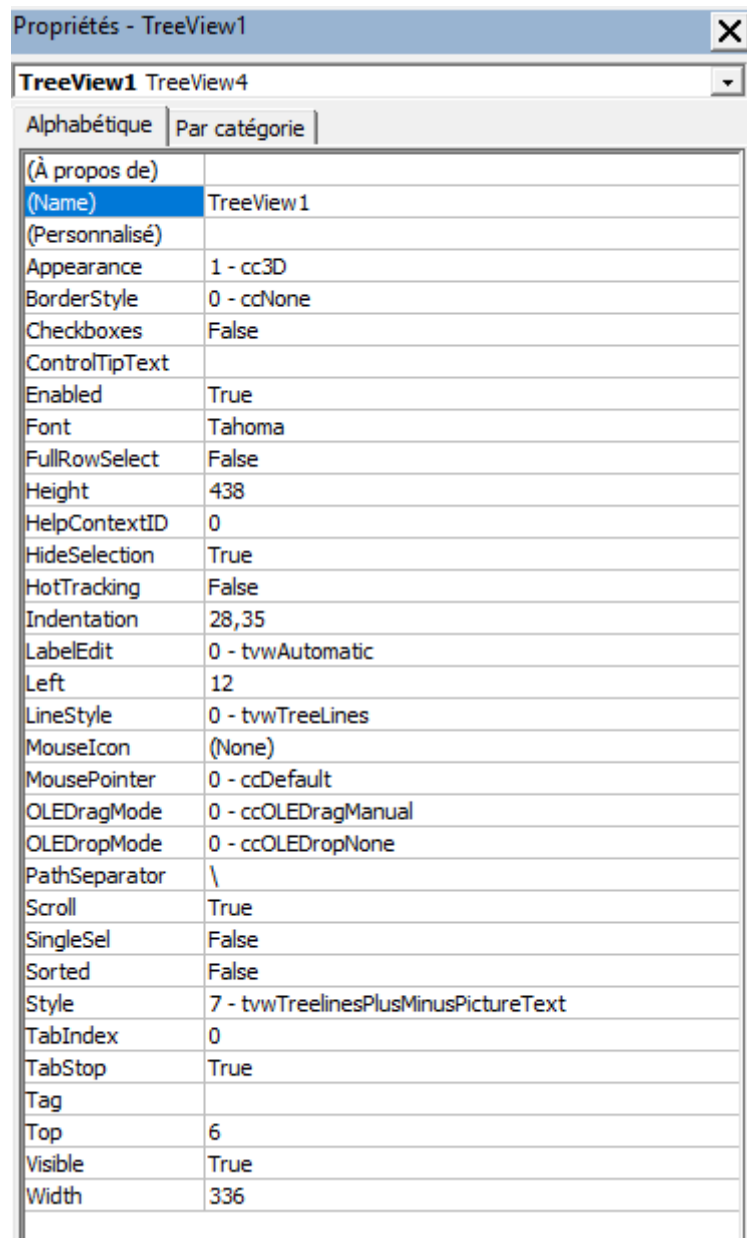
Présentation du TreeView

Un TreeView est un objet permettant de voir une présentation hiérarchique de ce que l'on veut.

Dans le fichier d'exemple, j'ai choisi de montrer l'arborescence de plusieurs sociétés et de leurs membres. Mais cela peut concerner des répertoires etc...

Dans un TreeView chaque élément est un nœud (Node). Il y a donc au moins un nœud Racine puis des nœuds enfants. Ces nœuds enfant peuvent devenir Parent etc etc. Lors de l'insertion de ceux-ci on peut définir par exemple que le Nœud B est l'enfant du Nœud A. Nous verrons cela plus tard.

Les paramètres du TreeView :



Description des paramètres types :

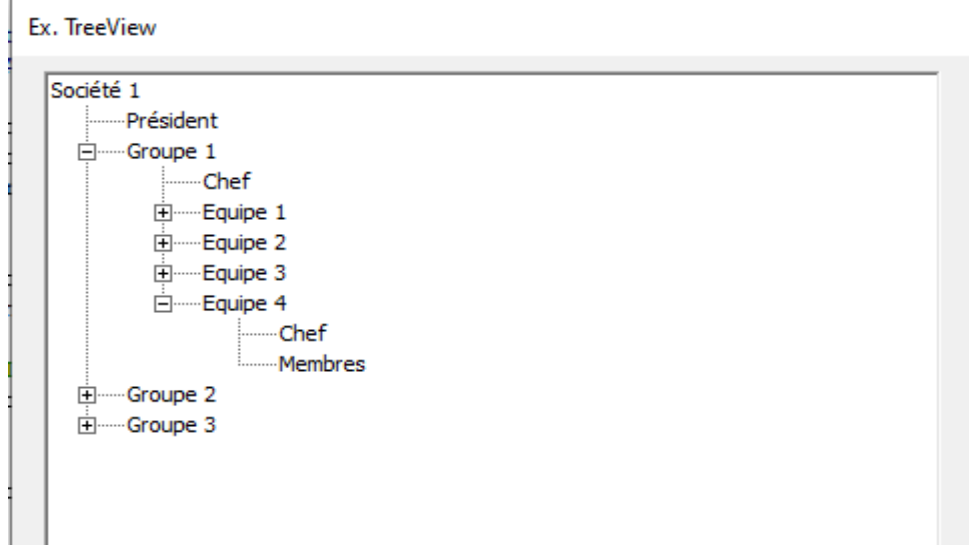
(Name) : Comme tout contrôle VBA, c'est lui qui nous permettra de le reconnaître entre tous.

CheckBoxes : Vous avez la possibilité de rajouter des checkboxes pour chaque Node. Pour lancer une macro dès que vous cochez un nœud vous pourrez utiliser la fonction : « Private Sub MaTreeView_NodeCheck(ByVal Node As MSComctlLib.Node) ».

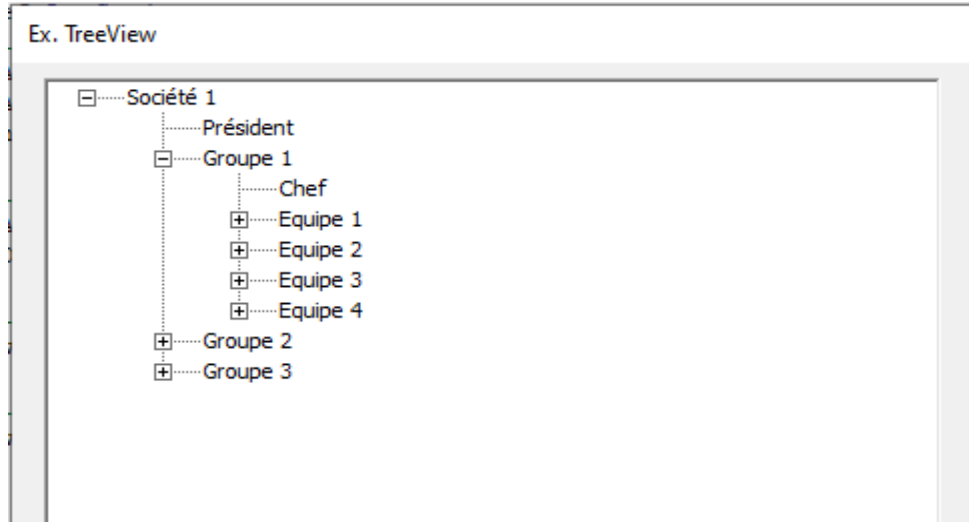
FullRowSelect : Option Graphique qui sélectionne toute la ligne du nœud qu'on a choisi.

LineStyle : 2 Possibilités :

- `tvwTreeLines`



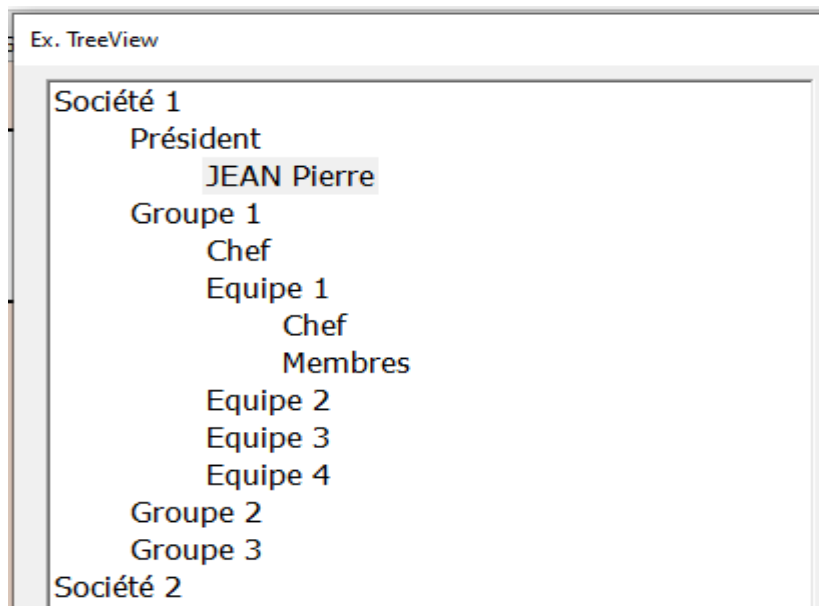
- `tvwRootLines`



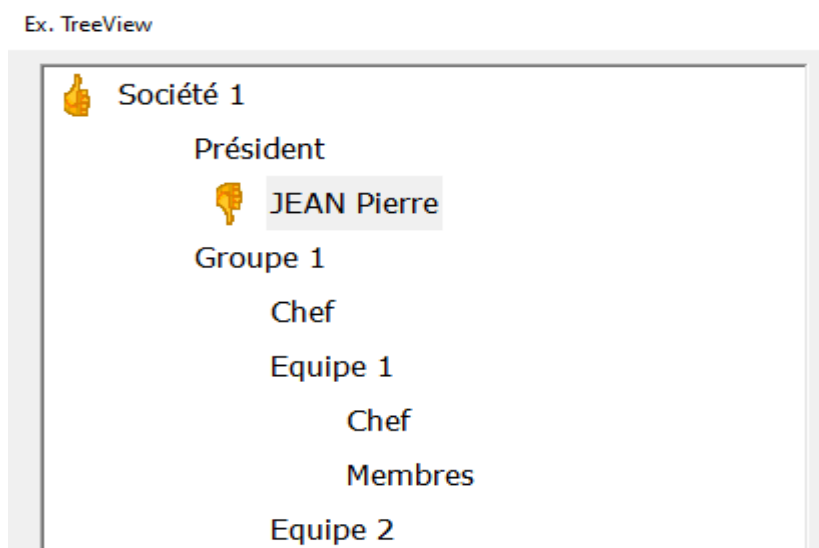
Globalement cela représente le fait de distinguer les nœuds racines. Visible uniquement si le Style est en Treelines Voir ci-dessous.

Style : 8 Possibilités :

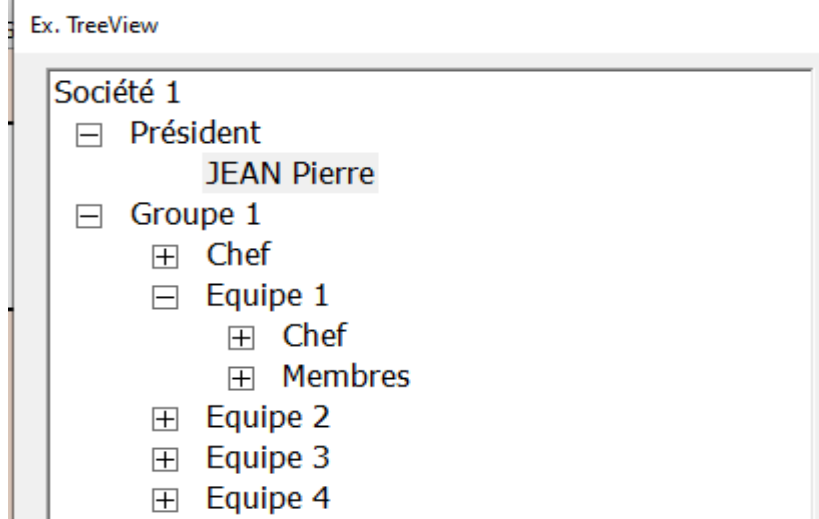
- `tvwTextOnly`



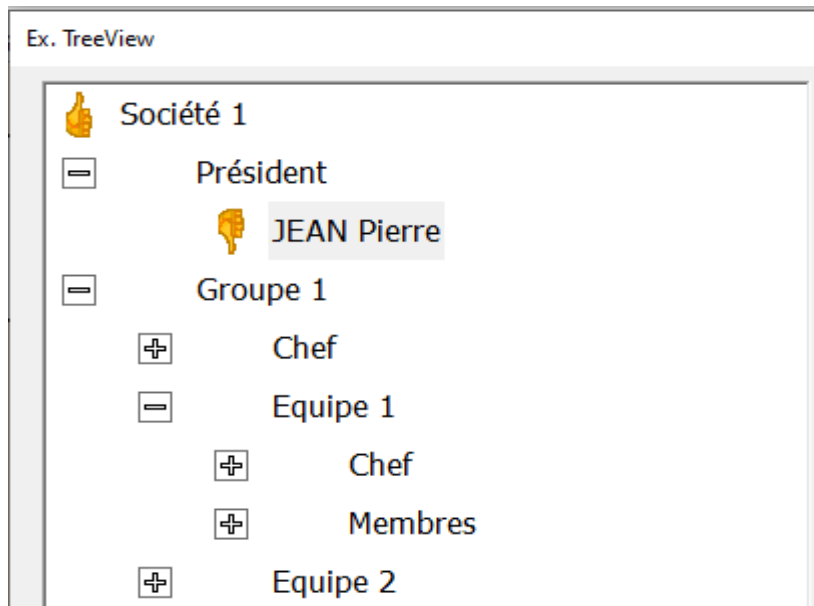
- `tvwPictureText`



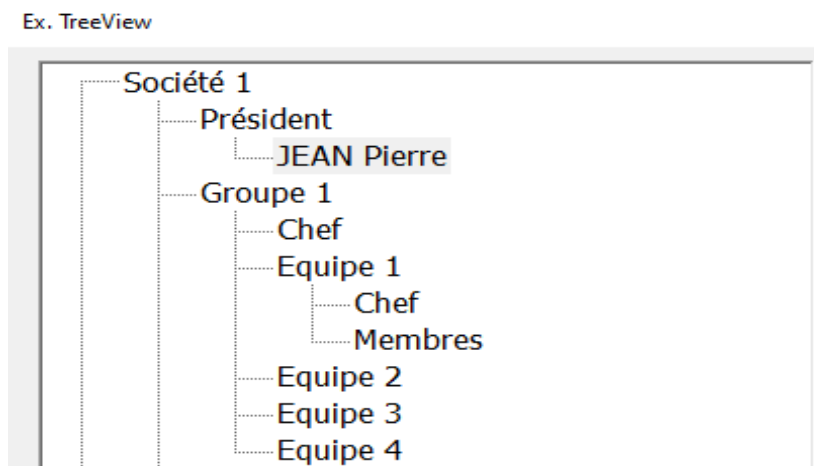
- `tvwPlusMinusText`



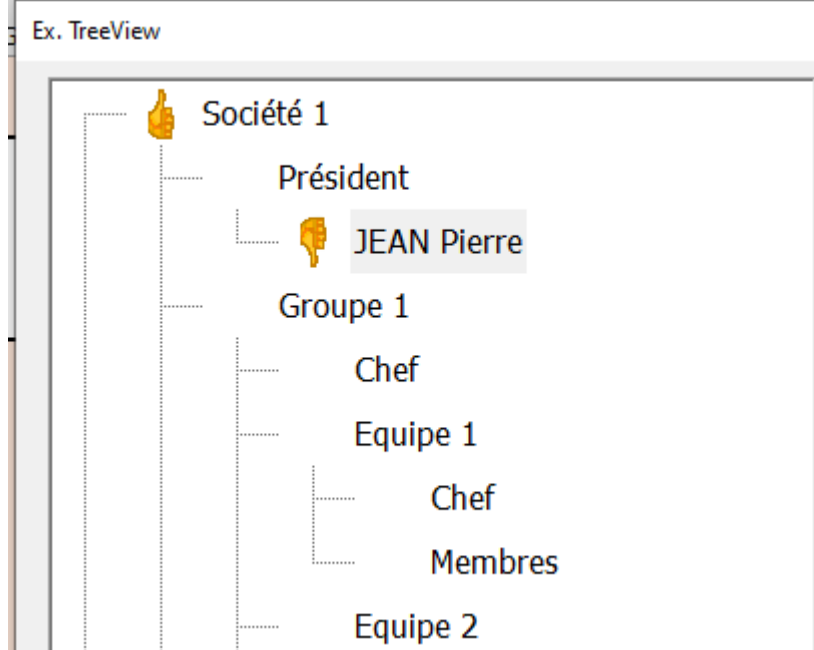
- `tvwPlusPictureText`



- `tvwTreelinesText`

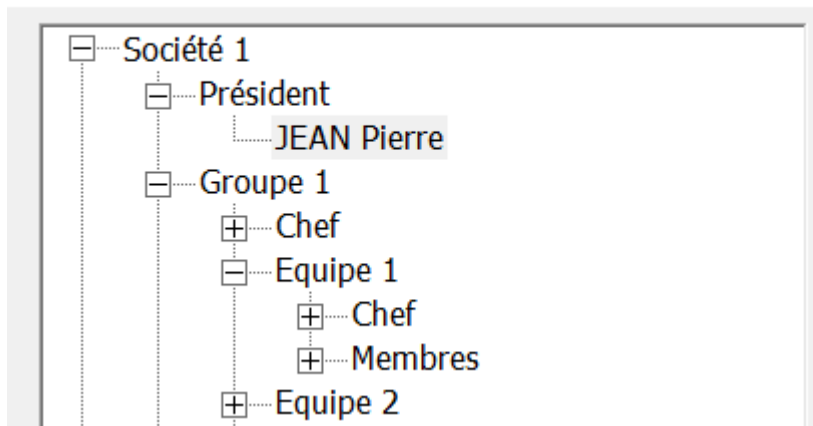


- `tvwTreelinesPictureText`



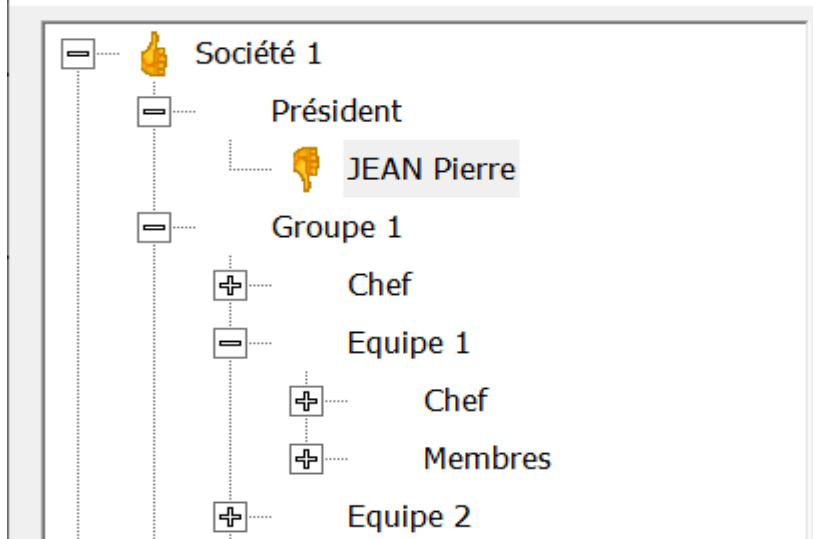
- `tvwTreelinesPlusMinusText`

Ex. TreeView



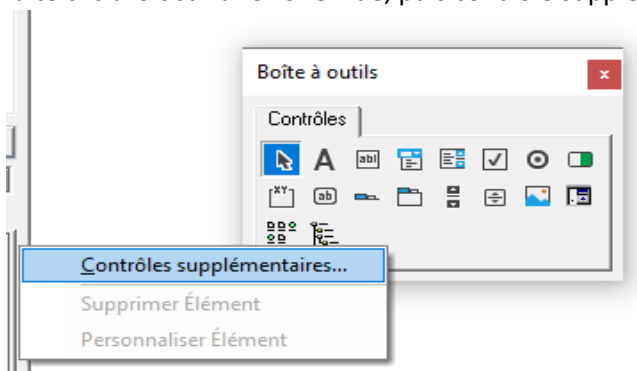
- `twvTreelinesPlusMinusPictureText`

Ex. TreeView

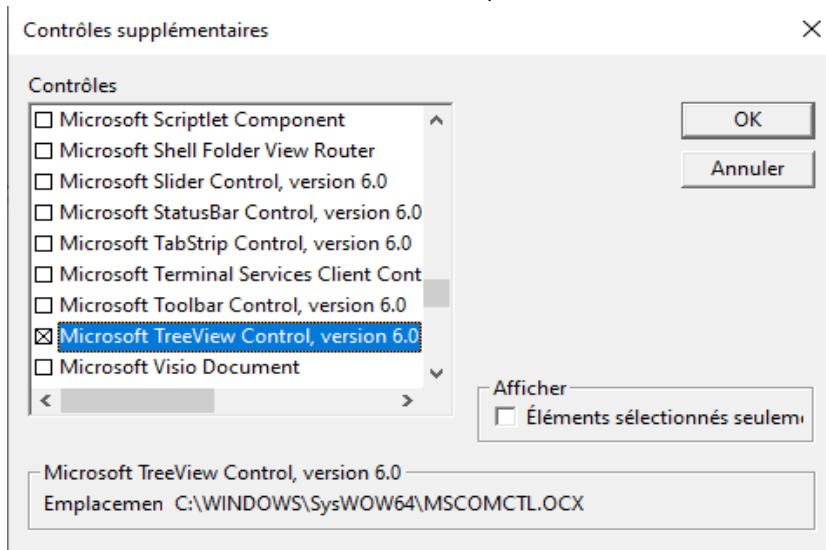


Ajouter le TreeView dans la boîte à outils

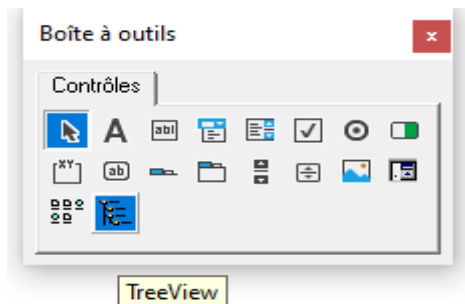
- Ouvrir Microsoft Visual Basic for Application :
 - o ALT + [F11]
 - o Onglet développeur, Visual Basic
- Créer un Userform :
 - o Dans la fenêtre Projet de Visual Basic (à gauche) faire clic droit, insertion, Userform
- Double cliquer sur le nouvel Userform pour rentrer dessus
- Il se peut que la fenêtre boîte à outils soit déjà présente sinon :
 - o Cliquer sur l'onglet Affichage puis Boîte à Outils.
- Sur la boîte à outils faire clic droit sur une zone vide, puis contrôle supplémentaire



- Descendre et Cocher : Microsoft TreeView Controls (A voir suivant votre version d'Excel)



- Cliquer sur OK.
- Voilà à quoi ressemble l'icône



Remplissage d'un TreeView

Commencer par ajouter un TreeView dans votre Userform et renommer le comme bon vous semble.

Dans mon exemple, le TreeView est renommé en MonTreeView.

Je vous conseille de bien préparer vos arborescences.

Vous pouvez les rentrer à la fin ou faire un programme pour les remplir pour vous. J'ai opté pour le programme. Moins encombrant et évolutif.

La Préparation :

J'ai choisi de dédié une feuille entière pour l'arborescence et une autre avec les données (Personnels) de mes sociétés.

Il existe une multitude de façon pour remplir un TreeView, vous trouverez la votre !

Mais si vous devez réutilisez ce programme et non vous en servir juste de cours je vous explique !

Dans la feuille Arborescence, La première ligne sert à placer tous les nœuds racines (Ici : Société 1, Société 2). Et ensuite dès la ligne 2, on définit les enfants de chaque nœud. Attention l'ordre est important. Comme vous pouvez le voir, les groupes et équipes portent le même nom. Dans un TreeView, une clef ne peut être utilisé que par un nœud et pour éviter de donner des clefs lambda et ne pas savoir les reconnaître, la clef de chacun est son chemin d'accès.

Dans mon programme pour séparer les nœuds racines vous pouvez laisser UNE ligne vierge.

Dans la feuille Données, on place simplement les informations de chacun.

Le fonctionnement :

Pour ajouter un nœud la commande est la suivante :

- `MonTreeView.Nodes.add` *Relative, Relationship, Key, Text, Image, SelectedImage*
 - *Relative* : (Facultatif) C'est ce que je nomme la ParentsKey. Elle est donc égale à la clef parent de ce nouveau nœud. Si c'est un nœud Racine, sa valeur doit être vide. Représente aussi l'index du nœud.
 - *Relationship* : (Facultatif) Plusieurs valeurs possibles :
 - *tvwLast* : Le nouveau nœud sera placé dans le même chemin que celui annoncé dans *Relative* mais en **dernière** position.
 - *tvwFirst* : Le nouveau nœud sera placé dans le même chemin que celui annoncé dans *Relative* mais en **première** position.
 - *tvwChild* : Le nouveau nœud sera enfant de celui annoncé dans *Relative*.
 - *tvwNext* : Le nouveau nœud sera placé juste **après** le nœud choisi dans *Relative*
 - *tvwPrevious* : Le nouveau nœud sera placé juste **avant** le nœud choisi dans *relative*
 - *Key* : (Facultatif) Représente la clef de ce nouveau nœud permettant de le retrouver.
 - *Text* : (Obligatoire) Texte que portera le nœud
 - *Image* : (Facultatif) Index d'une image. Permet de mettre une image sur le nœud. Nous verrons comment faire plus tard.
 - *SelectedImage* : (Facultatif) Index d'une image. Celle-ci sera mis si le nœud est sélectionné.

Le programme :

Le programme est visible et commenté en ouvrant Visual Basic.

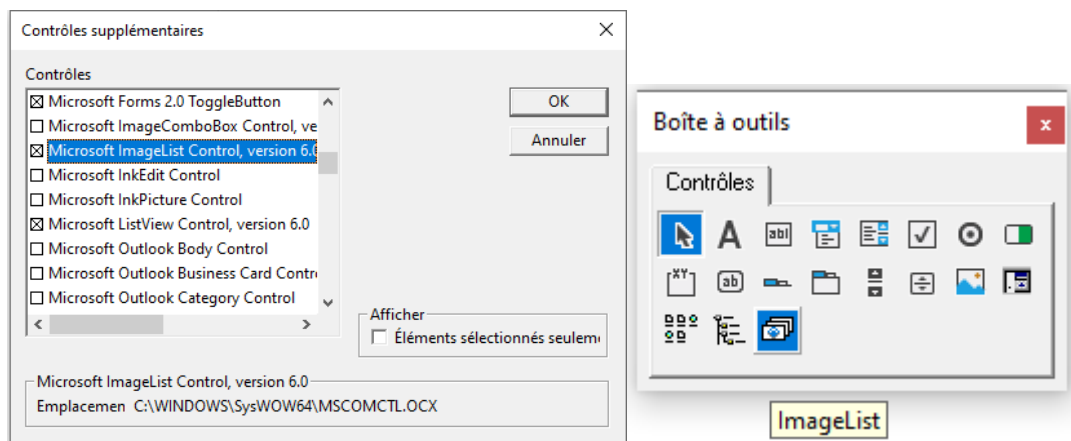
Ajouter une Image sur un nœud :

Pour ajouter une image sur dans un TreeView, nous devons passer par un autre contrôle : ImageList.

Ajouter ImageList à la boîte à outils :

Même manipulation que pour ajouter le TreeView mais cette fois cochez :

Microsoft ImageList Controls



Configurer l'ImageList :

Commencez par créer les variables de chemin d'accès aux Images Ex.

```
Dim Image1 as String
```

```
Dim Image2 as String
```

```
Image1 = ThisWorkbook.Path & "/MonImageA.jpg"
```

```
Image2 = ThisWorkbook.Path & "/MonImageB.jpg"
```

Nettoyer ImageList et définir la taille de vos images.

```
ImageList1.ListImages.Clear
```

```
ImageList1.ImageHeight = 32
```

```
ImageList1.ImagesWidth = 32
```

Chargez les Images dans le contrôle ImageList

```
ImageList1.ListImages.Add , "Clefdel'image1", LoadPicture(Image1)  
ImageList1.ListImages.Add , "Clefdel'image2", LoadPicture(Image2)
```

L'intégrer dans le nœud :

Vous devez dans un premier temps préciser à votre TreeView dans quel ImageList il va aller chercher ces images.

```
Set MonTreeView1.ImageList = ImageList1
```

Lors de l'ajout, il vous suffit donc de rentrer la clef de l'image voulu dans "Image" ou "SelectedImage"

```
MonTreeView.Nodes.Add , , Clef, Nom, "Clefdel'Image1"
```

Récupérer les informations d'un Nœud.

Un nœud sélectionné :

Pour un nœud sélectionné, on peut utiliser :

```
Private Sub MonTreeView_NodeClick(ByVal Node As MSComctlLib.Node)  
  
End sub
```

Pour les commandes voir ci-dessous, l'objet Node correspondra au nœud sélectionné.

Taper directement 'Node.'Paramètre''

Un nœud Coché

Pour un nœud sélectionné, on peut utiliser :

```
Private Sub MonTreeView_NodeCheck(ByVal Node As MSComctlLib.Node)
```

```
End sub
```

Pour les commandes voir ci-dessous, l'objet Node correspondra au nœud coché.

Taper directement 'Node.'Paramètre''

Globalement

Les propriétés des nœuds (MonTreeView.Nodes(index).Paramètre') :

- Parent : Renvoie le Parent de ce nœud. Si c'est un nœud Racine = Nothing
- Root : Renvoie le Nom du Nœud Racine d'où dépend Node
- Children : Renvoie le nombre d'objet enfant que possède Node
- Child : Renvoie le premier fils de Node
- Previous : Renvoie le nœud de même niveau hiérarchique précédent
- Next : Renvoie le nœud de même niveau hiérarchique suivant
- Expanded : Renvoie par True ou False si un nœud est développé ou non
- Index : Renvoie le numéro index du node (Le numéro dépend de l'ordre de création)
- Key : Renvoie la clef du node
- Text : Renvoie le texte du node